

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

E·XFI

Product Status	Active
Core Processor	dsPIC
Core Size	16-Bit
Speed	30 MIPs
Connectivity	CANbus, I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	132KB (44K x 24)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f6011a-30i-pt

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

	Access		Progra	m Space A	ddress	
Access Type	Space	<23>	<22:16>	<15>	<14:1>	<0>
Instruction Access	User	0		PC<22:1>		0
TBLRD/TBLWT	User (TBLPAG<7> = 0)	TBL	.PAG<7:0>		Data EA<15:0>	
TBLRD/TBLWT	Configuration (TBLPAG<7> = 1)	TBL	.PAG<7:0>		Data EA<15:0>	
Program Space Visibility	User	0	PSVPAG<	7:0>	Data EA<1	4:0>

FIGURE 3-3: DATA ACCESS FROM PROGRAM SPACE ADDRESS GENERATION



3.1.1 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

This architecture fetches 24-bit wide program memory. Consequently, instructions are always aligned. However, as the architecture is modified Harvard, data can also be present in program space.

There are two methods by which program space can be accessed: via special table instructions, or through the remapping of a 16K word program space page into the upper half of data space (see Section 3.1.2 "Data Access From Program Memory using Program Space Visibility"). The TBLRDL and TBLWTL instructions offer a direct method of reading or writing the lsw of any address within program space, without going through data space. The TBLRDH and TBLWTH instructions are the only method whereby the upper 8 bits of a program space word can be accessed as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to data space addresses. Program memory can thus be regarded as two 16-bit word wide address spaces, residing side by side, each with the same address range. TBLRDL and TBLWTL access the space which contains the Least Significant Data Word, and TBLRDH and TBLWTH access the space which contains the Most Significant Data Byte.

Figure 3-3 shows how the EA is created for table operations and data space accesses (PSV = 1). Here, P<23:0> refers to a program space word, whereas D<15:0> refers to a data space word. A set of table instructions are provided to move byte or word sized data to and from program space.

 TBLRDL: Table Read Low Word: Read the lsw of the program address; P<15:0> maps to D<15:0>.
Byte: Read one of the LSBs of the program address; P<7:0> maps to the destination byte when byte

P<7:0> maps to the destination byte when byte select = 0;

P < 15:8> maps to the destination byte when byte select = 1.

- TBLWTL: Table Write Low (refer to Section 6.0 "Flash Program Memory" for details on Flash Programming)
- TBLRDH: Table Read High Word: Read the most significant word of the program address; P<23:16> maps to D<7:0>; D<15:8> will always be = 0. Byte: Read one of the MSBs of the program

address;

P<23:16> maps to the destination byte when byte select = 0;

The destination byte will always be = 0 when byte select = 1.

 TBLWTH: Table Write High (refer to Section 6.0 "Flash Program Memory" for details on Flash Programming).



FIGURE 3-4: PROGRAM DATA TABLE ACCESS (LEAST SIGNIFICANT WORD)

4.1.3 MOVE AND ACCUMULATOR INSTRUCTIONS

Move instructions and the DSP accumulator class of instructions provide a greater degree of addressing flexibility than other instructions. In addition to the Addressing modes supported by most MCU instructions, move and accumulator instructions also support Register Indirect with Register Offset Addressing mode, also referred to as Register Indexed mode.

Note: For the MOV instructions, the addressing mode specified in the instruction can differ for the source and destination EA. However, the 4-bit Wb (register offset) field is shared between both source and destination (but typically only used by one).

In summary, the following Addressing modes are supported by move and accumulator instructions:

- Register Direct
- Register Indirect
- Register Indirect Post-modified
- · Register Indirect Pre-modified
- Register Indirect with Register Offset (Indexed)
- Register Indirect with Literal Offset
- 8-bit Literal
- 16-bit Literal
 - Note: Not all instructions support all the Addressing modes given above. Individual instructions may support different subsets of these Addressing modes.

4.1.4 MAC INSTRUCTIONS

The dual source operand DSP instructions (CLR, ED, EDAC, MAC, MPY, MPY.N, MOVSAC and MSC), also referred to as MAC instructions, utilize a simplified set of Addressing modes to allow the user to effectively manipulate the data pointers through register indirect tables.

The two source operand prefetch registers must be a member of the set {W8, W9, W10, W11}. For data reads, W8 and W9 will always be directed to the X RAGU and W10 and W11 will always be directed to the Y AGU. The effective addresses generated (before and after modification) must, therefore, be valid addresses within X data space for W8 and W9 and Y data space for W10 and W11.

Note: Register indirect with register offset addressing is only available for W9 (in X space) and W11 (in Y space). In summary, the following Addressing modes are supported by the ${\tt MAC}$ class of instructions:

- Register Indirect
- Register Indirect Post-modified by 2
- · Register Indirect Post-modified by 4
- · Register Indirect Post-modified by 6
- · Register Indirect with Register Offset (Indexed)

4.1.5 OTHER INSTRUCTIONS

Besides the various Addressing modes outlined above, some instructions use literal constants of various sizes. For example, BRA (branch) instructions use 16-bit signed literals to specify the branch destination directly, whereas the DISI instruction uses a 14-bit unsigned literal field. In some instructions, such as ADD Acc, the source of an operand or result is implied by the opcode itself. Certain operations, such as NOP, do not have any operands.

4.2 Modulo Addressing

Modulo Addressing is a method of providing an automated means to support circular data buffers using hardware. The objective is to remove the need for software to perform data address boundary checks when executing tightly looped code, as is typical in many DSP algorithms.

Modulo Addressing can operate in either data or program space (since the data pointer mechanism is essentially the same for both). One circular buffer can be supported in each of the X (which also provides the pointers into program space) and Y data spaces. Modulo Addressing can operate on any W register pointer. However, it is not advisable to use W14 or W15 for Modulo Addressing since these two registers are used as the Stack Frame Pointer and Stack Pointer, respectively.

In general, any particular circular buffer can only be configured to operate in one direction, as there are certain restrictions on the buffer start address (for incrementing buffers), or end address (for decrementing buffers) based upon the direction of the buffer.

The only exception to the usage restrictions is for buffers which have a power-of-2 length. As these buffers satisfy the start and end address criteria, they may operate in a Bidirectional mode (i.e., address boundary checks will be performed on both the lower and upper address boundaries).

4.2.3 MODULO ADDRESSING APPLICABILITY

Modulo Addressing can be applied to the Effective Address calculation associated with any W register. It is important to realize that the address boundaries check for addresses less than, or greater than the upper (for incrementing buffers), and lower (for decrementing buffers) boundary addresses (not just equal to). Address changes may, therefore, jump beyond boundaries and still be adjusted correctly.

Note: The modulo corrected Effective Address is written back to the register only when Pre-Modify or Post-Modify Addressing mode is used to compute the Effective Address. When an address offset (e.g., [W7+W2]) is used, modulo address correction is performed but the contents of the register remain unchanged.

4.3 Bit-Reversed Addressing

Bit-Reversed Addressing is intended to simplify data re-ordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which may be a constant value or register contents, is regarded as having its bit order reversed. The address source and destination are kept in normal order. Thus, the only operand requiring reversal is the modifier.

4.3.1 BIT-REVERSED ADDRESSING IMPLEMENTATION

Bit-Reversed Addressing is enabled when:

- 1. BWM (W register selection) in the MODCON register is any value other than '15' (the stack cannot be accessed using Bit-Reversed Addressing) **and**
- 2. the BREN bit is set in the XBREV register and
- 3. the Addressing mode used is Register Indirect with Pre-Increment or Post-Increment.

If the length of a bit-reversed buffer is $M = 2^N$ bytes, then the last 'N' bits of the data buffer start address must be zeros.

XB<14:0> is the bit-reversed address modifier or 'pivot point', which is typically a constant. In the case of an FFT computation, its value is equal to half of the FFT data buffer size.

Note:	All bit-reversed EA calculations assume
	word sized data (LSb of every EA is
	always clear). The XB value is scaled
	accordingly to generate compatible (byte)
	addresses.

When enabled, Bit-Reversed Addressing will only be executed for register indirect with pre-increment or post-increment addressing and word sized data writes. It will not function for any other addressing mode or for byte sized data, and normal addresses will be generated instead. When Bit-Reversed Addressing is active, the W address pointer will always be added to the address modifier (XB) and the offset associated with the Register Indirect Addressing mode will be ignored. In addition, as word sized data is a requirement, the LSb of the EA is ignored (and always clear).

Note:	Modulo Addressing and Bit-Reversed
	Addressing should not be enabled
	together. In the event that the user
	attempts to do this, Bit-Reversed Address-
	ing will assume priority when active for the
	X WAGU, and X WAGU Modulo Address-
	ing will be disabled. However, Modulo
	Addressing will continue to function in the X
	RAGU.

If Bit-Reversed Addressing has already been enabled by setting the BREN (XBREV<15>) bit, then a write to the XBREV register should not be immediately followed by an indirect read operation using the W register that has been designated as the bit-reversed pointer.

TABL	E 8-5:	PO	RTD RE	GISTE	R MAP	FOR d	sPIC30	F6011A	/6012A	(1)										
SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Res	et State	
TRISD	02D2	Ι			Ι	TRISD11	TRISD1(D TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	4 TRISD3	TRISD2	TRISD1	TRISD0	0000 111:	1 1111 1	111
PORTD	02D4	Ι	Ι	Ι	Ι	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	0000 0000	0 0000 0	000
LATD	02D6	Ι			Ι	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	0000 0000	0 0000 0	000
Legend Note	: 1: Ref	= uninitializ	zed bit; — = dsPIC30F F	= unimplem ⁻ <i>amily Ref</i> €	nented bit, r≀ ∍rence Man	ead as ' ₀ ' <i>ual"</i> (DS7(0046) for d	escriptions	of register	· bit fields.										
TABL	Е 8-6:	PO	RTD RE	GISTE	R MAP	FOR d	sPIC30	F6013A	/6014A	(1)										
SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Rese	et State	
TRISD	02D2	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1 1111 1	111
PORTD	02D4	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	0000 0000	0 0000 0	000
LATD	02D6	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	0000 0000	0 0000 0	000
Legend Note	: u 1: Ref	= uninitializ fer to the "	zed bit; — = dsPIC30F F	= unimplem ⁻ <i>amily Ref</i> €	าented bit, ห ร <i>rence Man</i>	ead as ' ₀ ' <i>ual"</i> (DS7(0046) for d	escriptions	of register	· bit fields.										
TABL	E 8-7:	PO	RTF RE	GISTE	R MAP I	⁼OR ds	PIC30	F6011A	/6012A	(1)										
SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset	t State	
TRISF	02DE		1	I		1				-	RISF6 TF	RISF5 T	RISF4	TRISF3	TRISF2	TRISF1	TRISF0	0000 0000	0111 11	11
PORTF	02E0	1	I	l		1	I		1	1	RF6	RF5	RF4	RF3	RF2	RF1	RFO	0000 0000	00 00 00	00
LATF	02E2	Ι				Ι				- F	ATF6 L	ATF5 L	ATF4	LATF3	LATF2	LATF1	LATF0	0000 0000	00 00 00	00
Legend Note	: u u	= uninitializ fer to the "(zed bit; — = dsPIC30F F	= unimplem ⁻ <i>amily Ref</i> ∈	าented bit, r ร <i>rence Man</i>	ead as ' _{0'} <i>ual"</i> (DS70	0046) for d	escriptions	of register	· bit fields.										
TABL	Е 8-8:	PO	RTF RE	GISTEI	R MAP I	⁼OR ds	PIC 301	F6013A	/6014A	(1)										
SFR Name	Addr.	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset	t State	
TRISF	02DE	Ι		Ι		Ι		- -	RISF8 T	RISF7 T	RISF6 TF	RISF5 T	RISF4	TRISF3	TRISF2	TRISF1	TRISF0	0000 0001	1111 11	11
PORTF	02E0					I		I	RF8	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	0000 0000	00 00 00	00
LATF	02E2							-	ATF8 L	ATF7 L	ATF6 L	ATF5 L	ATF4	LATF3	LATF2	LATF1	LATF0	0000 0000	00 00 00	00
Legend Note	:: -: Ret	= uninitializ fer to the "6	zed bit; — = dsPIC30F F	= unimplem ⁼amily Ref€	nented bit, r∉ ∍rence Man	ead as ' ₀ ' <i>ual"</i> (DS70	0046) for d	escriptions	of register	· bit fields.										

13.1 Timer2 and Timer3 Selection Mode

Each output compare channel can select between one of two 16-bit timers, Timer2 or Timer3.

The selection of the timers is controlled by the OCTSEL bit (OCxCON<3>). Timer2 is the default timer resource for the output compare module.

13.2 Simple Output Compare Match Mode

When control bits OCM<2:0> (OCxCON<2:0>) = 001, 010 or 011, the selected output compare channel is configured for one of three simple Output Compare Match modes:

- Compare forces I/O pin low
- Compare forces I/O pin high
- · Compare toggles I/O pin

The OCxR register is used in these modes. The OCxR register is loaded with a value and is compared to the selected incrementing timer count. When a compare occurs, one of these Compare Match modes occurs. If the counter resets to zero before reaching the value in OCxR, the state of the OCx pin remains unchanged.

13.3 Dual Output Compare Match Mode

When control bits OCM<2:0> (OCxCON<2:0>) = 100 or 101, the selected output compare channel is configured for one of two Dual Output Compare modes, which are:

- Single Output Pulse mode
- Continuous Output Pulse mode

13.3.1 SINGLE PULSE MODE

For the user to configure the module for the generation of a single output pulse, the following steps are required (assuming timer is off):

- 1. Determine instruction cycle time Tcy.
- Calculate desired pulse width value based on Tcy.
- 3. Calculate time to start pulse from timer start value of 0x0000.
- 4. Write pulse width start and stop times into OCxR and OCxRS Compare registers (x denotes channel 1, 2, ...,N).
- 5. Set Timer Period register to value equal to, or greater than value in OCxRS Compare register.
- 6. Set OCM<2:0> = 100.
- 7. Enable timer, TON (TxCON<15>) = 1.

To initiate another single pulse, issue another write to set OCM<2:0> = 100.

13.3.2 CONTINUOUS PULSE MODE

For the user to configure the module for the generation of a continuous stream of output pulses, the following steps are required:

- 1. Determine instruction cycle time Tcy.
- 2. Calculate desired pulse value based on Tcy.
- 3. Calculate timer to start pulse width from timer start value of 0x0000.
- 4. Write pulse width start and stop times into OCxR and OCxRS (x denotes channel 1, 2, ...,N) Compare registers, respectively.
- 5. Set Timer Period register to value equal to, or greater than value in OCxRS Compare register.
- 6. Set OCM<2:0> = 101.
- 7. Enable timer, TON (TxCON<15>) = 1.

13.4 Simple PWM Mode

When control bits OCM<2:0> (OCxCON<2:0>) = 110 or 111, the selected output compare channel is configured for the PWM mode of operation. When configured for the PWM mode of operation, OCxR is the main latch (read only) and OCxRS is the secondary latch. This enables glitchless PWM transitions.

The user must perform the following steps in order to configure the output compare module for PWM operation:

- 1. Set the PWM period by writing to the appropriate period register.
- 2. Set the PWM duty cycle by writing to the OCxRS register.
- 3. Configure the output compare module for PWM operation.
- 4. Set the TMRx prescale value and enable the Timer, TON (TxCON<15>) = 1.

13.4.1 INPUT PIN FAULT PROTECTION FOR PWM

When control bits OCM<2:0> (OCxCON<2:0>) = 111, the selected output compare channel is again configured for the PWM mode of operation with the additional feature of input Fault protection. While in this mode, if a logic '0' is detected on the OCFA/B pin, the respective PWM output pin is placed in the high-impedance input state. The OCFLT bit (OCxCON<4>) indicates whether a Fault condition has occurred. This state will be maintained until both of the following events have occurred:

- · The external Fault condition has been removed.
- The PWM mode has been re-enabled by writing to the appropriate control bits.

NOTES:

19.1 ADC Result Buffer

The module contains a 16-word dual port read only buffer, called ADCBUF0...ADCBUFF, to buffer the ADC results. The RAM is 12 bits wide but the data obtained is represented in one of four different 16-bit data formats. The contents of the sixteen ADC Result Buffer registers, ADCBUF0 through ADCBUFF, cannot be written by user software.

19.2 Conversion Operation

After the ADC module has been configured, the sample acquisition is started by setting the SAMP bit. Various sources, such as a programmable bit, timer time-outs and external events, will terminate acquisition and start a conversion. When the A/D conversion is complete, the result is loaded into ADCBUF0...ADCBUFF, and the DONE bit and the ADC interrupt flag ADIF are set after the number of samples specified by the SMPI bit. The ADC module can be configured for different interrupt rates as described in Section 19.3 "Selecting the Conversion Sequence".

Use the following steps to perform an Analog-to-Digital conversion:

- 1. Configure the ADC module:
 - a) Configure the analog pins, voltage reference and digital I/O.
 - b) Select the ADC input channels.
 - c) Select the ADC conversion clock.
 - d) Select the ADC conversion trigger.
 - e) Turn on the ADC module.
- 2. Configure ADC interrupt (if required):
 - a) Clear the ADIF bit.
 - b) Select the ADC interrupt priority.
- 3. Start sampling.
- 4. Wait the required acquisition time.
- 5. Trigger acquisition end, start conversion:
- 6. Wait for ADC conversion to complete, by either:
 - Waiting for the ADC interrupt, or
 - Waiting for the DONE bit to get set.
- 7. Read ADC result buffer, clear ADIF if required.

19.3 Selecting the Conversion Sequence

Several groups of control bits select the sequence in which the ADC connects inputs to the sample/hold channel, converts a channel, writes the buffer memory and generates interrupts.

The sequence is controlled by the sampling clocks.

The SMPI bits select the number of acquisition/ conversion sequences that would be performed before an interrupt occurs. This can vary from 1 sample per interrupt to 16 samples per interrupt.

The BUFM bit will split the 16-word results buffer into two 8-word groups. Writing to the 8-word buffers will be alternated on each interrupt event.

Use of the BUFM bit will depend on how much time is available for the moving of the buffers after the interrupt.

If the processor can quickly unload a full buffer within the time it takes to acquire and convert one channel, the BUFM bit can be '0' and up to 16 conversions (corresponding to the 16 input channels) may be done per interrupt. The processor will have one acquisition and conversion time to move the sixteen conversions.

If the processor cannot unload the buffer within the acquisition and conversion time, the BUFM bit should be '1'. For example, if SMPI<3:0> (ADCON2<5:2>) = 0111, then eight conversions will be loaded into 1/2 of the buffer, following which an interrupt occurs. The next eight conversions will be loaded into the other 1/2 of the buffer. The processor will have the entire time between interrupts to move the eight conversions.

The ALTS bit can be used to alternate the inputs selected during the sampling sequence. The input multiplexer has two sets of sample inputs: MUX A and MUX B. If the ALTS bit is '0', only the MUX A inputs are selected for sampling. If the ALTS bit is '1' and SMPI<3:0> = 0000 on the first sample/convert sequence, the MUX A inputs are selected and, on the next acquire/convert sequence, the MUX B inputs are selected.

The CSCNA bit (ADCON2<10>) will allow the multiplexer input to be alternately scanned across a selected number of analog inputs for the MUX A group. The inputs are selected by the ADCSSL register. If a particular bit in the ADCSSL register is '1', the corresponding input is selected. The inputs are always scanned from lower to higher numbered inputs, starting after each interrupt. If the number of inputs selected is greater than the number of samples taken per interrupt, the higher numbered inputs are unused. The following figure depicts the recommended circuit for the conversion rates above 100 ksps. The dsPIC30F6014A is shown as an example.



The configuration procedures below give the required setup values for the conversion speeds above 100 ksps.

19.7.1 200 KSPS CONFIGURATION GUIDELINE

The following configuration items are required to achieve a 200 ksps conversion rate.

- · Comply with conditions provided in Table 19-2.
- Connect external VREF+ and VREF- pins following the recommended circuit as shown in Figure 19-2.
- Set SSRC<2.0> = 111 in the ADCON1 register to enable the auto convert option.
- Enable automatic sampling by setting the ASAM control bit in the ADCON1 register.
- Write the SMPI<3.0> control bits in the ADCON2 register for the desired number of conversions between interrupts.

Configure the ADC clock period to be:
1

by writing to the ADCS<5:0> control bits in the ADCON3 register.

• Configure the sampling time to be 1 TAD by writing: SAMC<4:0> = 00001.

The following figure shows the timing diagram of the ADC running at 200 ksps. The TAD selection in conjunction with the guidelines described above allows a conversion speed of 200 ksps. See Example 19-1 for code example.

20.2.2 OSCILLATOR START-UP TIMER (OST)

In order to ensure that a crystal oscillator (or ceramic resonator) has started and stabilized, an Oscillator Start-up Timer is included. It is a simple 10-bit counter that counts 1024 Tosc cycles before releasing the oscillator clock to the rest of the system. The time-out period is designated as TOST. The TOST time is involved every time the oscillator has to restart (i.e., on POR, BOR and wake-up from Sleep). The Oscillator Start-up Timer is applied to the LP, XT, XTL and HS Oscillator modes (upon wake-up from Sleep, POR and BOR) for the primary oscillator.

20.2.3 LP OSCILLATOR CONTROL

Enabling the LP oscillator is controlled with two elements:

- The current oscillator group bits COSC<2:0>
- The LPOSCEN bit (OSCCON register)

The LP oscillator is ON (even during Sleep mode) if LPOSCEN = 1. The LP oscillator is the device clock if:

- COSC<2:0> = 000 (LP selected as main oscillator) and
- LPOSCEN = 1

Keeping the LP oscillator ON at all times allows for a fast switch to the 32 kHz system clock for lower power operation. Returning to the faster main oscillator will still require a start-up time.

20.2.4 PHASE LOCKED LOOP (PLL)

The PLL multiplies the clock which is generated by the primary oscillator. The PLL is selectable to have either gains of x4, x8 and x16. Input and output frequency ranges are summarized in Table 20-3.

TABLE 20-3: PLL FREQUENCY RANGE

Fin	PLL Multiplier	Fout
4 MHz-10 MHz	x4	16 MHz-40 MHz
4 MHz-10 MHz	x8	32 MHz-80 MHz
4 MHz-7.5 MHz	x16	64 MHz-120 MHz

The PLL features a lock output, which is asserted when the PLL enters a phase locked state. Should the loop fall out of lock (e.g., due to noise), the lock signal will be rescinded. The state of this signal is reflected in the read-only LOCK bit in the OSCCON register.

20.2.5 FAST RC OSCILLATOR (FRC)

The FRC oscillator is a fast (7.37 MHz $\pm 2\%$ nominal) internal RC oscillator. This oscillator is intended to provide reasonable device operating speeds without the use of an external crystal, ceramic resonator or RC network. The FRC oscillator can be used with the PLL to obtain higher clock frequencies.

The dsPIC30F operates from the FRC oscillator whenever the current oscillator selection control bits in the OSCCON register (OSCCON<14:12>) are set to '001'.

The 6-bit field specified by TUN<3:0> (OSCTUN<3:0>) allows the user to tune the internal fast RC oscillator (nominal 7.37 MHz). The user can tune the FRC oscillator within a range of $\pm 6\%$ in steps of 0.75% around the factory-calibrated setting (see Table 20-4).

Note:	OSCTUN functionality has been provided
	to help customers compensate for
	temperature effects on the FRC frequency
	over a wide range of temperatures. The
	tuning step size is an approximation and is
	neither characterized nor tested.

If OSCCON<14:12> are set to '111' and FPR<4:0> are set to '00101', '00110' or '00111', then a PLL multiplier of 4, 8 or 16 (respectively) is applied.

Note:	When a 16x PLL is used, the FRC oscilla-
	tor must not be tuned to a frequency
	greater than 7.5 MHz.

TABLE 20-4: FRC TUNING

TUN<3:0> Bits	FRC Frequency
0111	+5.25%
0110	+4.50%
0101	+3.75%
0100	+3.00%
0011	+2.25%
0010	+1.50%
0001	+0.75%
0000	Center Frequency (oscillator is
	running at calibrated frequency)
1111	-0.75%
1110	-1.50%
1101	-2.25%
1100	-3.00%
1011	-3.75%
1010	-4.50%
1001	-5.25%
1000	-6.00%

20.4.1.1 POR with Long Crystal Start-up Time (with FSCM Enabled)

The oscillator start-up circuitry is not linked to the POR circuitry. Some crystal circuits (especially low frequency crystals) will have a relatively long start-up time. Therefore, one or more of the following conditions is possible after the POR timer and the PWRT have expired:

- · The oscillator circuit has not begun to oscillate.
- The Oscillator Start-up Timer has NOT expired (if a crystal oscillator is used).
- The PLL has not achieved a LOCK (if PLL is used).

If the FSCM is enabled and one of the above conditions is true, then a clock failure trap will occur. The device will automatically switch to the FRC oscillator and the user can switch to the desired crystal oscillator in the trap ISR.

20.4.1.2 Operating without FSCM and PWRT

If the FSCM is disabled and the Power-up Timer (PWRT) is also disabled, then the device will exit rapidly from Reset on power-up. If the clock source is FRC, LPRC, EXTRC or EC, it will be active immediately.

If the FSCM is disabled and the system clock has not started, the device will be in a frozen state at the Reset vector until the system clock starts. From the user's perspective, the device will appear to be in Reset until a system clock is available.

20.4.2 BOR: PROGRAMMABLE BROWN-OUT RESET

The BOR (Brown-out Reset) module is based on an internal voltage reference circuit. The main purpose of the BOR module is to generate a device Reset when a brown-out condition occurs. Brown-out conditions are generally caused by glitches on the AC mains (i.e., missing portions of the AC cycle waveform due to bad power transmission lines or voltage sags due to excessive current draw when a large inductive load is turned on).

The BOR module allows selection of one of the following voltage trip points:

- 2.6V-2.71V
- 4.1V-4.4V
- 4.58V-4.73V

Note: The BOR voltage trip points indicated here are nominal values provided for design guidance only.

A BOR will generate a Reset pulse which will reset the device. The BOR will select the clock source, based on the device Configuration bit values (FOS<2:0> and FPR<4:0>). Furthermore, if an oscillator mode is selected, the BOR will activate the Oscillator Start-up Timer (OST). The system clock is held until OST expires. If the PLL is used, then the clock will be held until the LOCK bit (OSCCON<5>) is '1'.

Concurrently, the POR time-out (TPOR) and the PWRT time-out (TPWRT) will be applied before the internal Reset is released. If TPWRT = 0 and a crystal oscillator is being used, then a nominal delay of TFSCM = 100 μ s is applied. The total delay in this case is (TPOR + TFSCM).

The BOR status bit (RCON<1>) will be set to indicate that a BOR has occurred. The BOR circuit, if enabled, will continue to operate while in Sleep or Idle modes and will reset the device should VDD fall below the BOR threshold voltage.

FIGURE 20-6:

EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



- Note 1: External Power-on Reset circuit is required only if the VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.
 - 2: R should be suitably chosen so as to make sure that the voltage drop across R does not violate the device's electrical specification.
 - R1 should be suitably chosen so as to limit any current flowing into MCLR from external capacitor C, in the event of MCLR/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

Note: Dedicated supervisory devices, such as the MCP1XX and MCP8XX, may also be used as an external Power-on Reset circuit.

20.5 Watchdog Timer (WDT)

20.5.1 WATCHDOG TIMER OPERATION

The primary function of the Watchdog Timer (WDT) is to reset the processor in the event of a software malfunction. The WDT is a free running timer, which runs off an on-chip RC oscillator, requiring no external component. Therefore, the WDT timer will continue to operate even if the main processor clock (e.g., the crystal oscillator) fails.

20.5.2 ENABLING AND DISABLING THE WDT

The Watchdog Timer can be "Enabled" or "Disabled" only through a Configuration bit (FWDTEN) in the Configuration register FWDT.

Setting FWDTEN = 1 enables the Watchdog Timer. The enabling is done when programming the device. By default, after chip-erase, FWDTEN bit = 1. Any device programmer capable of programming dsPIC30F devices allows programming of this and other Configuration bits.

If enabled, the WDT will increment until it overflows or "times out". A WDT time-out will force a device Reset (except during Sleep). To prevent a WDT time-out, the user must clear the Watchdog Timer using a CLRWDT instruction.

If a WDT times out during Sleep, the device will wakeup. The WDTO bit in the RCON register will be cleared to indicate a wake-up resulting from a WDT time-out.

Setting FWDTEN = 0 allows user software to enable/ disable the Watchdog Timer via the SWDTEN control bit (RCON<5>).

20.6 Low-Voltage Detect

The Low-Voltage Detect (LVD) module is used to detect when the VDD of the device drops below a threshold value, VLVD, which is determined by the LVDL<3:0> bits (RCON<11:8>) and is thus user programmable. The internal voltage reference circuitry requires a nominal amount of time to stabilize, and the BGST bit (RCON<13>) indicates when the voltage reference has stabilized.

In some devices, the LVD threshold voltage may be applied externally on the LVDIN pin.

The LVD module is enabled by setting the LVDEN bit (RCON<12>).

20.7 Power-Saving Modes

There are two power-saving states that can be entered through the execution of a special instruction, ${\tt PWRSAV}.$

These are: Sleep and Idle.

The format of the PWRSAV instruction is as follows:

PWRSAV <parameter>,

where:

'parameter' defines Idle or Sleep mode.

20.7.1 SLEEP MODE

In Sleep mode, the clock to the CPU and peripherals is shutdown. If an on-chip oscillator is being used, it is shutdown.

The Fail-Safe Clock Monitor is not functional during Sleep, since there is no clock to monitor. However, LPRC clock remains active if WDT is operational during Sleep.

The Brown-out protection circuit, if enabled, will remain functional during Sleep.

The processor wakes up from Sleep if at least one of the following conditions has occurred:

- On any interrupt that is individually enabled and meets the required priority level
- On any Reset (POR, BOR and MCLR)
- On WDT time-out

On waking up from Sleep mode, the processor will restart the same clock that was active prior to entry into Sleep mode. When clock switching is enabled, bits COSC<2:0> will determine the oscillator source that will be used on wake-up. If clock switch is disabled, then there is only one system clock.

Note: If a POR or BOR occurred, the selection of the oscillator is based on the FOS<2:0> and FPR<4:0> Configuration bits.

If the clock source is an oscillator, the clock to the device is held off until OST times out (indicating a stable oscillator). If PLL is used, the system clock is held off until LOCK = 1 (indicating that the PLL is stable). Either way, TPOR, TLOCK and TPWRT delays are applied.

If EC, FRC, LPRC or EXTRC oscillators are used, then a delay of TPOR (~ 10 μ s) is applied. This is the smallest delay possible on wake-up from Sleep.

Moreover, if LP oscillator was active during Sleep, and LP is the oscillator used on wake-up, then the start-up delay will be equal to TPOR. PWRT delay and OST timer delay are not applied. In order to have the smallest possible start-up delay when waking up from Sleep, one of these faster wake-up options should be selected before entering Sleep.

Base Instr #	Assembly Mnemonic		Assembly Syntax	Description	# of Words	# of Cycles	Status Flags Affected
1	ADD	ADD	Acc	Add Accumulators	1	1	OA.OB.SA.SB
		ADD	f	f = f + WREG	1	1	C,DC,N,OV,Z
		ADD	f,WREG	WREG = f + WREG	1	1	C,DC,N,OV,Z
		ADD	#lit10,Wn	Wd = lit10 + Wd	1	1	C,DC,N,OV,Z
		ADD	Wb,Ws,Wd	Wd = Wb + Ws	1	1	C,DC,N,OV,Z
		ADD	Wb,#lit5,Wd	Wd = Wb + lit5	1	1	C,DC,N,OV,Z
		ADD	Wso,#Slit4,Acc	16-bit Signed Add to Accumulator	1	1	OA,OB,SA,SB
2	ADDC	ADDC	f	f = f + WREG + (C)	1	1	C,DC,N,OV,Z
		ADDC	f,WREG	WREG = $f + WREG + (C)$	1	1	C,DC,N,OV,Z
		ADDC	#lit10,Wn	Wd = lit10 + Wd + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,Ws,Wd	Wd = Wb + Ws + (C)	1	1	C,DC,N,OV,Z
		ADDC	Wb,#lit5,Wd	Wd = Wb + lit5 + (C)	1	1	C,DC,N,OV,Z
3	AND	AND	f	f = f .AND. WREG	1	1	N,Z
		AND	f,WREG	WREG = f .AND. WREG	1	1	N,Z
		AND	#lit10,Wn	Wd = lit10 .AND. Wd	1	1	N,Z
		AND	Wb,Ws,Wd	Wd = Wb .AND. Ws	1	1	N,Z
		AND	Wb,#lit5,Wd	Wd = Wb .AND. lit5	1	1	N,Z
4	ASR	ASR	f	f = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	f,WREG	WREG = Arithmetic Right Shift f	1	1	C,N,OV,Z
		ASR	Ws,Wd	Wd = Arithmetic Right Shift Ws	1	1	C,N,OV,Z
		ASR	Wb,Wns,Wnd	Wnd = Arithmetic Right Shift Wb by Wns	1	1	N,Z
		ASR	Wb,#lit5,Wnd	Wnd = Arithmetic Right Shift Wb by lit5	1	1	N,Z
5	BCLR	BCLR	f,#bit4	Bit Clear f	1	1	None
		BCLR	Ws,#bit4	Bit Clear Ws	1	1	None
6	BRA	BRA	C,Expr	Branch if Carry	1	1 (2)	None
		BRA	GE,Expr	Branch if greater than or equal	1	1 (2)	None
		BRA	GEU,Expr	Branch if unsigned greater than or equal	1	1 (2)	None
		BRA	GT,Expr	Branch if greater than	1	1 (2)	None
		BRA	GTU,Expr	Branch if unsigned greater than	1	1 (2)	None
		BRA	LE,Expr	Branch if less than or equal	1	1 (2)	None
		BRA	LEU,Expr	Branch if unsigned less than or equal	1	1 (2)	None
		BRA	LT,Expr	Branch if less than	1	1 (2)	None
		BRA	LTU,Expr	Branch if unsigned less than	1	1 (2)	None
		BRA	N,Expr	Branch if Negative	1	1 (2)	None
		BRA	NC,Expr	Branch if Not Carry	1	1 (2)	None
		BRA	NN, Expr	Branch if Not Negative	1	1 (2)	None
		BRA	NOV, Expr	Branch if Not Overflow	1	1 (2)	None
		BRA	NZ,Expr	Branch if Not Zero	1	1 (2)	None
		BRA	OA,Expr	Branch if Accumulator A overflow	1	1 (2)	None
		BRA	OB,Expr	Branch if Accumulator B overflow	1	1 (2)	None
		BRA	OV,Expr	Branch if Overflow	1	1 (2)	None
		BRA	SA,Expr	Branch if Accumulator A saturated	1	1 (2)	None
		BRA	SB,Expr	Branch if Accumulator B saturated	1	1 (2)	None
		BRA	Expr	Branch Unconditionally	1	2	None
		BRA	Z,Expr	Branch if Zero	1	1 (2)	None
		BRA	Wn	Computed Branch	1	2	None
7	BSET	BSET	f,#bit4	Bit Set f	1	1	None
		BSET	Ws,#bit4	Bit Set Ws	1	1	None
8	BSW	BSW.C	Ws,Wb	Write C bit to Ws <wb></wb>	1	1	None
		BSW.Z	Ws,Wb	Write Z bit to Ws <wb></wb>	1	1	None

TABLE 21-2: INSTRUCTION SET OVERVIEW



TABLE 23-26: INPUT CAPTURE TIMING REQUIREMENTS

АС СНА	RACTER	ISTICS	Standard Operation (unless otherwise Operating temper	Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended					
Param No.	Symbol	Characte	ristic ⁽¹⁾	Min	Мах	Units	Conditions		
IC10	TccL	ICx Input Low Time	No Prescaler	0.5 Tcy + 20		ns			
			With Prescaler	10		ns			
IC11	TccH	ICx Input High Time	No Prescaler	0.5 Tcy + 20	-	ns			
			With Prescaler	10	_	ns			
IC15	TccP	ICx Input Period		(2 Tcy + 40)/N		ns	N = prescale value (1, 4, 16)		

Note 1: These parameters are characterized but not tested in manufacturing.

FIGURE 23-10: OUTPUT COMPARE MODULE (OCx) TIMING CHARACTERISTICS



TABLE 23-27: OUTPUT COMPARE MODULE TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic ⁽¹⁾	Min	Тур ⁽²⁾	Мах	Units	Conditions
OC10	TccF	OCx Output Fall Time	—	—	_	ns	See Parameter DO32
OC11	TccR	OCx Output Rise Time	_	_	_	ns	See Parameter DO31

Note 1: These parameters are characterized but not tested in manufacturing.

2: Data in "Typ" column is at 5V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.





FIGURE 23-17: SPI MODULE SLAVE MODE (CKE = 1) TIMING CHARACTERISTICS







64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



	Units			MILLIMETERS		
Dime	nsion Limits	MIN	NOM	MAX		
Number of Leads	Ν	64				
Lead Pitch	е	0.50 BSC				
Overall Height	А	-	-	1.20		
Molded Package Thickness	A2	0.95	1.00	1.05		
Standoff	A1	0.05	-	0.15		
Foot Length	L	0.45	0.60	0.75		
Footprint	L1	1.00 REF				
Foot Angle	φ	0°	3.5°	7°		
Overall Width	E	12.00 BSC				
Overall Length	D	12.00 BSC				
Molded Package Width	E1	10.00 BSC				
Molded Package Length	D1	10.00 BSC				
Lead Thickness	С	0.09	_	0.20		
Lead Width	b	0.17	0.22	0.27		
Mold Draft Angle Top	α	11°	12°	13°		
Mold Draft Angle Bottom	β	11°	12°	13°		

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.

2. Chamfers at corners are optional; size may vary.

3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.

- 4. Dimensioning and tolerancing per ASME Y14.5M.
 - BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B

Revision E (February 2011)

This revision includes minor typographical and formatting changes throughout the data sheet text.

The major changes are referenced by their respective section in Table A-1.

TABLE A-1: MAJOR SECTION UPDATES

Section Name	Update Description
Section 20.0 "System Integration"	Added a shaded note on OSCTUN functionality in Section 20.2.5 " Fast RC Oscillator (FRC) ".
Section 23.0 "Electrical Characteristics"	Updated the maximum MIPS for the Operating MIPS vs. Voltage VDD range of 3.0-3.6V for dsPIC30F601XA-20I and dsPIC30F601XA-30I devices (see Table 23-1).
	Added Operating Current (IDD) parameters DC27a and DC27b (see Table 23-5).
	Added Idle Current (IIDLE) parameters DC47a and DC47b (see Table 23-6).
	Updated the maximum value for parameter DI19 and the minimum value for parameter DI29 in the I/O Pin Input Specifications (see Table 23-8).
	Removed parameter D136 and updated the minimum, typical, maximum, and conditions for parameters D122 and D134 in the Program and EEPROM specifications (see Table 23-12).

Alignment (Figure)	35
Effect of Invalid Memory Accesses (Table)	34
MCI and DSP (MAC Class) Instructions Example	34
	5 04
Метогу Мар	31
Memory Map for dsPIC30F6011A/6013A	32
Memory Map for dsPIC30F6012A/6014A	33
Noar Data Space	35
Software Stack	35
Spaces	31
Width	34
Data Convertor Interface (DCI) Medule	100
	123
Data EEPROM Memory	57
Erasing	58
Frasing Block	58
Erasing, word	58
Protection Against Spurious Write	60
Reading	57
Write Verify	60
Writing	59
Writing, Block	60
Writing Word	59
DC Characteristics	174
	1/4
Brown-out Reset18	31, 182
I/O Pin Input Specifications	180
I/O Pin Output Specifications	180
	100
	177
Low-Voltage Detect	180
LVDL	181
Operating Current (100)	176
	170
Power-Down Current (IPD)	178
Program and EEPROM	182
DCI Module	
Bit Clock Generator	127
	127
Buffer Alignment with Data Frames	129
Buffer Control	123
Buffer Data Alignment	123
Buffer Length Control	120
	123
COFS Pin	123
CSCK Pin	123
CSDI Pin	123
CSDO Modo Dit	120
	130
CSDO Pin	123
Data Justification Control Bit	128
Device Frequencies for Common Codec CSCK Fr	equen-
	107
	121
Digital Loopback Mode	130
Enable	125
Frame Sync Generator	125
France Our Made Ocentral Dite	125
	120
Frame Sync Mode Control Bits	123
I/O Pins	
I/O Pins I/C Pins	130
I/O Pins I/O Pins Interrupts Introduction	130
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction	130
I/O Pins I/O Pins Interrupts Introduction Master Frame Sync Operation	130 123 125
I/O Pins I/O Pins Interrupts Introduction Master Frame Sync Operation Operation	130 123 125 125
Frame Sync Mode Control Bits I/O Pins Introduction Master Frame Sync Operation Operation Operation During CPU Idle Mode	130 123 125 125 130
Prame Sync Mode Control Bits I/O Pins Introduction Master Frame Sync Operation Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode	130 123 125 125 130 130
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Popoing Slot Enable Pite	130 123 125 125 130 130
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits	130 123 125 125 130 130 128
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits	130 123 125 125 130 130 128 129
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map	130 123 125 125 130 130 128 129 132
Frame Sync Mode Control Bits I/O Pins Interrupts Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit	130 123 125 125 130 130 128 129 132 128
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit Slave Frame Surg Operation	130 123 125 125 130 130 130 128 129 132 132
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit Slave Frame Sync Operation	130 123 125 125 130 130 128 129 122 122 122
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit Slave Frame Sync Operation Slot Enable Bits Operation with Frame Sync	130 123 125 125 130 130 130 128 129 132 128 128 126 128
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit Slave Frame Sync Operation Slot Enable Bits Operation with Frame Sync Slot Status Bits	130 123 125 125 130 130 128 129 128 128 128 126 128 128 128
Frame Sync Mode Control Bits I/O Pins Interrupts Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Slot Enable Bits Register Map Sample Clock Edge Control Bit Slave Frame Sync Operation Slot Enable Bits Operation with Frame Sync Slot Status Bits Synchronous Data Transfers	130 123 125 125 130 130 128 129 128
Frame Sync Mode Control Bits I/O Pins Interrupts Introduction Master Frame Sync Operation Operation During CPU Idle Mode Operation During CPU Sleep Mode Receive Slot Enable Bits Receive Status Bits Register Map Sample Clock Edge Control Bit Slave Frame Sync Operation Slot Enable Bits Operation with Frame Sync Slot Status Bits Synchronous Data Transfers	130 123 125 125 130 130 128 129 128 128 126 128 128 120 128 128

AC-Link Mode	197
Multichannel, I ² S Modes	195
Timing Requirements	
AC-Link Mode	197
Multichannel J ² S Modes	196
Transmit Slot Enable Rits	128
Transmit Status Bits	120
Transmit/Deceive Shift Degister	123
Haderflow Mede Centrel Bit	120
Word Size Selection Dite	105
Word Size Selection Bits	125
Development Support	169
Device Configuration	
Register Map	160
Device Configuration Registers	158
FBORPOR	158
FBS	158
FGS	158
FOSC	158
FSS	158
FWDT	158
Device Overview	91
Disabling the LIART	105
Divide Support	20
Instructions (Table)	20
DSP Engino	
DOF LINGING	
Nulliplier	
Dual Output Compare Match Mode	86
Continuous Pulse Mode	86
Single Pulse Mode	86
F	
E	
E Electrical Characteristics	173
E Electrical Characteristics	173 183
E Electrical Characteristics AC DC	173 183 174
E Electrical Characteristics AC DC Enabling and Setting Up UART	173 183 174
E Electrical Characteristics AC DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections	173 183 174 105
E Electrical Characteristics	173 183 174 105 105
E Electrical Characteristics	173 183 174 105 105
E Electrical Characteristics	173 183 174 105 105 135
E Electrical Characteristics	173 183 174 105 105 135 107
E Electrical Characteristics	173 183 174 105 105 105 135 107 127
E Electrical Characteristics	173 183 174 105 105 135 107 127 127
E Electrical Characteristics	173 183 174 105 105 135 137 127 125 100
E Electrical Characteristics ACDC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud Rate Bit Clock Frequency COFSG Period Serial Clock Rate Time Quantum for Clock Congration	173 183 174 105 105 135 135 107 127 125 100
E Electrical Characteristics ACDC DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud Rate	173 183 174 105 105 135 135 137 127 125 100 117
E Electrical Characteristics AC. DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections. Enabling the UART Equations ADC Conversion Clock Baud Rate. Bit Clock Frequency. COFSG Period. Serial Clock Rate Time Quantum for Clock Generation. Errata	173 183 174 105 105 105 135 107 127 125 100 117 9
E Electrical Characteristics	173 183 174 105 105 135 135 137 127 125 100 117 9
E Electrical Characteristics	173 183 174 105 105 105 135 107 127 125 100 117 9 191
E Electrical Characteristics	173 183 174 105 105 105 135 107 127 125 100 117 9 191 184
E Electrical Characteristics	173 183 174 105 105 105 135 107 127 125 100 117 9 191 184 191
E Electrical Characteristics	173 183 174 105 105 105 107 127 125 100 117 9 191 184 191 192
E Electrical Characteristics	173 183 174 105 105 105 105 107 127 125 100 117 9 191 191 192 192
E Electrical Characteristics AC DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud Rate Bit Clock Frequency COFSG Period Serial Clock Rate Time Quantum for Clock Generation Errata External Clock Timing Characteristics Type A, B and C Timer External Clock Timing Requirements Type A Timer Type B Timer Type C Timer External Interrupt Requests	173 183 174 105 105 105 105 107 127 125 100 117 9 191 191 191 192 192 49
E Electrical Characteristics ACDC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud RateBit Clock FrequencyCOFSG PeriodSerial Clock RateTime Quantum for Clock GenerationErrata External Clock Timing Characteristics Type A, B and C TimerExternal Clock Timing Requirements Type A TimerType B TimerType B TimerType C TimerExternal Interrupt Requests	173 183 174 105 105 135 107 127 125 100 117 9 191 191 192 192 192 49
E Electrical Characteristics ACDC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud RateBit Clock FrequencyCOFSG PeriodSerial Clock Rate Time Quantum for Clock GenerationErrata External Clock Timing Characteristics Type A, B and C Timer External Clock Timing Requirements Type A Timer Type A Timer Type C Timer External Interrupt Requests F	173 183 174 105 105 105 135 107 127 125 100 117 9 191 191 192 192 192 49
E Electrical Characteristics ACDC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud RateBit Clock FrequencyCOFSG PeriodSerial Clock Rate Time Quantum for Clock GenerationErrata External Clock Timing Characteristics Type A, B and C Timer External Clock Timing Requirements Type A TimerType B Timer	173 183 174 105 105 105 135 107 127 125 100 117 9 191 191 192 192 49 49
E Electrical Characteristics ACDC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections Enabling the UART Equations ADC Conversion Clock Baud RateBit Clock FrequencyCOFSG PeriodSerial Clock Rate Time Quantum for Clock Generation Serial Clock Rate Time Quantum for Clock Generation ErrataExternal Clock Timing Characteristics Type A, B and C Timer External Clock Timing Requirements Type A Timer Type B Timer Type C Timer External Interrupt Requests	173 183 174 105 105 105 135 107 127 125 100 117 9 191 191 192 192 49 49 51
E Electrical Characteristics AC. DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections. Enabling the UART Equations ADC Conversion Clock Baud Rate. Bit Clock Frequency. COFSG Period. Serial Clock Rate Time Quantum for Clock Generation Errata External Clock Timing Characteristics Type A, B and C Timer. External Clock Timing Requirements Type A Timer. Type A Timer. Type C Timer. External Interrupt Requests. F Fast Context Saving Flash Program Memory Control Registers.	173 183 174 105 105 105 105 107 127 125 100 117 9 191 191 192 192 192 49 51 52
E Electrical Characteristics AC. DC Enabling and Setting Up UART Setting Up Data, Parity and Stop Bit Selections. Enabling the UART Equations ADC Conversion Clock Baud Rate. Bit Clock Frequency. COFSG Period. Serial Clock Rate Time Quantum for Clock Generation Errata. External Clock Timing Characteristics Type A, B and C Timer. External Clock Timing Requirements Type A Timer. Type A Timer. Type B Timer. Type C Timer. External Interrupt Requests. F Fast Context Saving Flash Program Memory Control Registers. NVMADR	173 183 174 105 105 105 105 107 127 127 125 100 117 9 191 191 192 192 192 49 49 52 52

I

I/O Pin Specifications	
Input	180
Output	180